

Reference 1

Japanese Patent Application Public-disclosure No. 1-255933
Japanese Patent Application Public-disclosure date: October 12, 1989

Title of the invention: Sweep out control method
Japanese Patent Application No. 63-84616
Japanese Patent Application date: April 6, 1988

[Embodiment]

Next, an embodiment of the present invention will be explained in detail.

First, a circuit on which the embodiment is based will be briefly described. Referring to Fig. 1, for processing to be performed at instruction fetch (IF) stage are provided; instruction cache I Cache for storing an instruction read from main storage 8; address register AIC for supplying an address to the instruction cache I Cache; incrementer + 1 for updating the content of the address register AIC; instruction buffer IB for temporarily storing an instruction from the instruction cache I Cache; and selector 21 for selecting either an instruction from the instruction cache I Cache or an instruction from the instruction buffer IB.

Operand address creation (AC) stage requires; instruction register IR 20 for storing a modified instruction selected by and output from the selector 21; registers BR and GR for storing an address from the instruction register IR 20 or an address to be indexed by a value of the address; and address operator AD-ADDER for calculating values from the instruction register IR 20 and registers BR and GR to thereby generate an address.

At address translation (AT) stage, logical address register LAR for storing a logical address from the address operator AD-ADDER and address translation buffer TLB for reading a physical address in response to the logical address are provided.

Next, for processing to be performed at cache access (CA)

stage are provided: physical address register PAR for storing a physical address from the address translation buffer TLB; operand cache 0 Cache for storing an operand from the main storage 8 at a location designated by address information from the physical address register PAR and reading the thus stored operand; registers BR and GR; and selector 22 for selecting the contents from the registers BR and GR.

For execution (EX) stage are provided: operator FLT-ADD for performing computations; shifter SHT for shifting; execution register EXR for storing the operation results, operation intermediate results and operands from the FLT-ADD, shifter SHT and operand cache 0 Cache; register REGISTER for storing the content from the selector 22 and operation results and operation intermediate results from the address operator FLT-ADD and shifter SHT; and arithmetic control means 1 to which information from the execution register EXR and register REGISTER are input and from which an instruction to update the instruction counter 15 is output to the line 101.

Result storing (ST) stage includes; read data register RDR 5 for storing the operation results from the operator FLT-ADD and shifter SHT; write address register WA2 for setting a write address; incrementer 3 for incrementing the write address from the register WA2 by +1; register WF4 for setting an instruction termination flag provided via the line 102; instruction termination flag buffer WFB 6 for storing the flag from the register WF4 at a location designated by the write address register WA2; store buffer 7 for storing the content of the read data register 5 provided via the line 103 at a location represented by the address from the register WA2; read address register 9 for storing the read address corresponding to the store buffer 7; incrementer 10 for updating the content of the read address register 9; memory access control means 11 for outputting to the line 105 an instruction signal for controlling buffer sweep sent out from the store buffer 7 to the main storage 8; register MAR 12 for sending out the content from the store buffer 7 to the main storage 8 via the line 104

in response to the instruction signal supplied from the memory access control means 11 via the line 105; instruction counter (hereafter referred to as IC) update control circuit 13 for outputting to the line 102 "1" as an instruction termination flag in response to the instruction counter update instruction signal supplied from the operation control means 1 via the line 101 when updating the instruction counter, and for otherwise outputting "0" to the line 102; instruction counter 16; register IC 15 for setting a value from the instruction counter 16 in response to a signal provided from the IC update control circuit 13 via the line 102; register WFI 17 for storing a signal from the IC update control circuit 13; selection control circuit 18 for selecting either an output from the register WFI 17 or an output from the instruction termination flag buffer WFB 6; and store buffer sweep out pointer 19 for selecting an output from the incrementer 23, an output from the decrementer 24 or an output from the memory access control means 11 and storing the selected output.

Next, an operation of the embodiment of the present invention will be specifically described with reference to Figs. 1 ~ 3. Referring to Figs. 1 and 2, in the cycle 1, instruction A from the main storage 8 is stored in the instruction cache I Cache. The instruction A is a four-word instruction. Next, in the cycle 2, instruction B from the main storage 8 is stored in the instruction cache I Cache while the instruction A is sent from the instruction cache I Cache via the selector 21 and instruction register IR 20 to the adder AD-ADDER where an address is generated. The instructions A, B and C are processed in the aforementioned manner. The instruction B is a two-word instruction and the instruction C is a one-word instruction.

Referring to Figs. 1 ~ 3, the result of an operation on the first word of the instruction A that was stored in the read data register RDR 5 in the cycle 6, is stored at the address 1 in the store buffer 7 via the line 103 in the cycle 7.

Since an IC update instruction is not being output via the line 101, a value "0" stored in the register WF0 4 from the

IC update control circuit 13 via the line 102 is stored in a corresponding word in the instruction termination flag buffer 6 while the result of an operation on the first word of the instruction A is stored in the store buffer 7.

In the cycle 10, the content of the instruction counter register 15 changes from "A" to "B", and "1" is stored in the instruction termination flag buffer 4, and the incrementer 23 operates to change the content of the store buffer sweep out pointer 19 from "0" to "1". "1" in this case means that the store buffer 7 contains data equivalent to one instruction that can be swept out.

Since the store buffer 7 contains no data that can be swept out when a value of the store buffer sweep out pointer 19 is "0", the memory access control means 11 provides the register 12 with an instruction signal for stopping a sweep out operation from the store buffer 7 to the main storage 8. As the content of the sweep out pointer 19 is currently "1", the memory access control means 11 instructs the register 12 to supply the content of the store buffer 7 to the main storage 8 via the line 104. In response to the instruction, the register 12 starts sweeping out instruction units A-1, A-2, A-3 and A-4 of the instruction A from the store buffer 7 to the main storage 8.

In the cycle 10, storage of the fourth word of the instruction A in the store buffer 7 is completed and in the next cycle 11, an operation for storing the first word of the instruction B in the store buffer 7 is started.

While the content of the instruction counter register 15 is updated in the cycle 12, "1" is stored in the instruction termination flag buffer 6 in synchronization with the updating operation. Since an operation of sweeping out the third word of the instruction A from the store buffer 7 to the main storage 8 is being performed, instruction units B-1 and B-2 of the instruction B cannot be swept out to the main storage 8. Thus, a value of the store buffer sweep out pointer 19 is incremented from "1" to "2", which means that the store buffer 7 contains data indicating two instructions that can be swept out.

In the cycle 13, the instruction counter 16 updates the content of the instruction counter register 15 from "2" to "3" and the result of an operation on the instruction C is stored, in synchronization with which a value "1" is stored in the instruction termination flag buffer 6. Since the fourth word of the instruction A is being swept out from the store buffer 7 to the main storage 8 at this time, the result of an operation on the instruction C cannot be swept out to the main storage 8.

Thus, a value of the store buffer sweep out pointer 19 is counted up from "2" to "3", which indicates that the store buffer 7 contains stored data indicating three instructions that can be swept out.

In the cycle 13, the result of an operation on the fourth word of the instruction A is swept out to the main storage 8 and "1" is output from the instruction termination flag buffer 6. Then, in the cycle 14, the content of the store buffer sweep out pointer 19 is decremented by the decrementer 24 and as a result, the content of the pointer 19 changes from "3" to "2". The value "2" means that the store buffer 7 contains stored data indicating two instructions that can be swept out.

Also in the cycle 14, a sweep out operation of the result of an operation on the first word of the instruction B is started, and in the cycle 15, an instruction unit of the second word of the instruction B is swept out to the main storage 8. When "1" from the instruction termination flag buffer 6 is detected, a value of the store buffer sweep out pointer 19 is decremented to "1" by the decrementer 24. This means that the store buffer 7 contains stored data indicating one instruction that can be swept out.

When the instruction C is swept out of the store buffer 7 in the cycle 16, "1" is detected in the instruction termination flag buffer 6, and in the cycle 17, a value of the store buffer sweep out pointer 19 is counted down to "0" by the decrementer 24, which means that the store buffer 7 contains no data that can be swept out.

The instruction termination buffer 6 has a configuration illustrated in Fig. 3, and when a value of the instruction counter register 15 is changed with such timing that store data is written to the store buffer 7, "1" is written. Otherwise, "0" is written.

Further, if a flag to be stored in the instruction termination flag buffer 6 is "1", a value of the store buffer sweep out pointer 19 is counted up. On the other hand, if the flag is "0", the value is not counted up and the current value of the pointer 19 is maintained.

Further, while a sweep out operation from the store buffer 7 to the main storage 8 is conducted, a read operation of the instruction termination flag buffer 6 is conducted. If a value of the read instruction termination flag is "1", a value of the store buffer sweep out pointer 19 is counted down, whereas if a value of the instruction termination flag is "0", the value is not counted down and the current value of the sweep out pointer 19 is maintained.

Next, an operation to be performed when a failure is detected at the execution stage will be described. For example, if a failure is detected at the execution stage EX in the cycle 7 of the instruction A, update of the instruction counter 16, storage in instruction termination flag buffer 6, update of the store buffer sweep out pointer 19 and sweep out of the instruction A from the store buffer 7 to the main storage 8 are all inhibited and a retry of the instruction A is instructed.

[Effect of the invention]

The present system adds an instruction termination flag to an instruction unit in response to the store request issued in the store instruction to thereby control such that rewrite of the content of the main storage 8 is delayed until the instruction ends.

Further, as the present system is provided with the store buffer sweep out pointer 19, the oldest instruction unit stored in the store buffer 7 can be first swept out to the main storage

8, thereby advantageously improving a retry rate.

[Brief explanation of the drawings]

Fig. 1 is an illustration of an embodiment of the present invention.

Fig. 2 is a time chart describing an operation of the embodiment of the present invention.

Fig. 3 illustrates a relationship between a store buffer, an instruction termination flag buffer and a store buffer sweep out pointer.

Fig. 4 is an illustration describing pipeline processing.

Fig. 5 is a time chart describing an example of a prior art.

[Description of the reference numerals]

- 1: operation control means
- 2: write address register
- 3: read data register
- 6: instruction termination flag buffer
- 7: store buffer
- 8: main storage
- 9: read address register
- 11: memory access control means
- 13: instruction counter update control circuit
- 15: instruction counter register
- 16: instruction counter
- 18: selection control circuit
- 19: store buffer sweep out pointer
- 20: instruction register

⑫ 公開特許公報(A) 平1-255933

⑤Int.Cl.⁴G 06 F 9/38
12/08

識別記号

3 5 0

庁内整理番号

X-7361-5B
C-7010-5B

④公開 平成1年(1989)10月12日

審査請求 未請求 請求項の数 1 (全7頁)

⑥発明の名称 掃出し制御方式

②特 願 昭63-84616

②出 願 昭63(1988)4月6日

⑦発 明 者 宮 沢 文 彦 東京都港区芝5丁目33番1号 日本電気株式会社内

⑧出 願 人 日本電気株式会社 東京都港区芝5丁目33番1号

⑨代 理 人 弁理士 柳 川 信

明 細 書

1. 発明の名称

掃出し制御方式

2. 特許請求の範囲

(1) 複数回のストア動作で一命令関連の情報を格納するストアバッファ手段と、このストアバッファ手段に格納される一命令関連の情報の最終回のストア動作に反応して値を増加させるか減少させ、前記ストアバッファ手段からの掃出し動作に反応して値を減少させるか増加させるストアバッファ掃出しポインタ手段と、このストアバッファ掃出しポインタ手段の値が前記ストアバッファ手段からの掃出し対象となる命令関連の情報がないことを示しているとき前記ストアバッファ手段からの掃出し動作を抑止する抑止手段とを含むことを特徴とする掃出し制御方式。

3. 発明の詳細な説明

技術分野

本発明は掃出し制御方式に関し、特に1回以上

のストアを必要とする命令の演算処理において、ストアバッファから主記憶装置への掃出し制御方式に関する。

従来技術

従来のパイプライン処理型情報処理装置でのパイプライン処理は次のようにして行なわれる。第4図を参照すると、このパイプライン処理の一例では、アドレス手段により命令キャッシュから命令を取出す命令取出(IF)ステージ、このステージで取出された命令を命令レジスタに格納したあとこの命令のオペランドにもとづきアドレス加算器で論理アドレスを生成するオペランドアドレス(AL)ステージ、このステージで生成された論理アドレスを論理アドレスレジスタに格納したあとアドレス変換バッファで論理アドレスを物理アドレスに変換するアドレス変換(AT)ステージ、このステージで変換された物理アドレスを物理アドレスレジスタに格納したあと、この物理アドレスでオペランドキャッシュをアクセスしオペランドを読出すオペランドキャッシュアクセス(CA)ス

テージ、このステージで読出されたオペランドを
実行レジスタに格納したあと演算器で演算する演
算実行(EX)ステージ、およびこの(EX)ステー
ジで演算された結果を格納する結果格納(ST)ス
テージの6つのステージに分割されている。

第5図を参照すると、従来この種のパイプライ
ン処理型情報処理装置では、一命令が全てストア
バッファに格納し終るまで主記憶装置への格納が
抑止されていた。すなわち4バイトである命令A
は、タイミング10で命令Aの4バイト目が格納
されてから次のタイミング11で命令Aの1バイ
ト目が掃出される。また2バイト命令である命令
Bのストアバッファから主記憶装置への掃出しは、
命令Aの全てが掃出された後、タイミング15で
行なわれることになる。

よって、命令Aのように一命令で多くのストア
処理をしなければならない命令では、命令リトラ
イ不可期間が長くなるという欠点がある。

発明の目的

本発明の目的は上述の欠点を除去するようにし

た掃出し制御方式を提供することにある。

発明の構成

本発明による掃出し制御方式は、複数回のストア
動作で一命令関連の情報を格納するストアバッ
ファ手段と、このストアバッファ手段に格納され
る一命令関連の情報の最終回のストア動作に
て値を増加させるか減少させ、前記ストアバッ
ファ手段からの掃出し動作に
て値を減少させるか増加させるストアバッファ掃出しポイン
タ手段と、このストアバッファ掃出しポイン
タ手段の値が前記ストアバッファ手段からの掃出し対象
となる命令関連の情報が
ないことを示しているとき前記ストアバッファ手段からの掃出し動作を抑
止する抑止手段とを含む構成である。

実施例

次に本発明の一実施例について詳細に説明する。

まず、本発明の一実施例の前提となる回路につ
いて簡単に説明する。第1図を参照すると、命令
取出し(IF)ステージの処理のため主記憶装置8
から読出された命令を格納する命令キャッシュ1

Cache、この命令キャッシュ1 Cacheにアドレス
を供給するアドレスレジスタAIC、このアドレス
レジスタAICの内容を更新するためのインクリメ
ンタ+1、命令キャッシュ1 Cacheからの命令を
一旦格納するための命令バッファ1Bおよび命令
キャッシュ1 Cacheおよび命令バッファ1Bのうち
どちらか一方からの命令を選択するためのセレクト
2 1を備えている。

オペランドアドレス生成(AC)ステージでは、
セレクト2 1で選択出力された変更後の命令を格
納する命令レジスタ1R20、この命令レジスタ1R20
からのアドレスまたはその値により索引されるア
ドレスを格納するためのレジスタBRおよびGR、お
よびこの命令レジスタ1R20、レジスタBRおよびGR
からの値を計算しアドレスを生成するためのアド
レス演算器AD.ADDERが必要である。

アドレス変換(AT)ステージのためには、アド
レス演算器AD.ADDERから与えられる論理アドレス
を格納する論理アドレスレジスタ1ARおよびこの
論理アドレスに
て物理アドレスを
読出すた

めのアドレス変換バッファ1LBが用意されている。

次にキャッシュアクセス(CA)ステージの処理
のためアドレス変換バッファ1LBから与えられる
物理アドレスを格納するための物理アドレスレジ
スタPAR、この物理アドレスレジスタPARからの
アドレス情報で指示される位置に主記憶装置8か
らのオペランドを格納し、格納されたオペランド
を
読出すためのオペランドキャッシュ0 Cache、
レジスタ群BRおよびGR、更にこれらレジスタ群BR
およびGRからの内容を選択するためのセレクト2
2が用意されている。

演算実行(EX)ステージのためには、演算を行
うための演算器FLT.ADD、桁合わせを行うための
シフタSHI、これら演算器FLT.ADD、シフタSHI
およびオペランドキャッシュ0 Cacheからの演算
結果、演算中間結果およびオペランドを格納する
ための実行レジスタEXR、セレクト2 2からの内
容および演算器FLT.ADDアドレスおよびシフタ
SHIからの演算結果、演算中間結果を格納するた

めのレジスタREGISTER、実行レジスタEXR およびレジスタREGISTERからの情報を入力し、命令カウンタ15の更新指示を線101に出力する演算制御部1が設けられている。

結果格納(ST)ステージでは、演算器FLI.ADDおよびシフトSHTからの演算結果を格納するためのリードデータレジスタRDR 5の他、書込みアドレスをセットする書込みアドレスレジスタWA2、このレジスタWA2からの書込みアドレスを+1増加させるためのインクリメント3、線102を介して与えられる命令終了フラグをセットするレジスタWF4、このレジスタWF4からのフラグを書込みアドレスレジスタWA2で指示される位置に格納する命令終了フラグバッファWFB 6、線103を介して与えられるリードデータレジスタ5の内容を、レジスタWA2からのアドレスで指示される位置に格納するためのストアバッファ7、このストアバッファ7に対する読出アドレスを格納するための読出アドレスレジスタ9、この読出アドレスレジスタ9の内容を更新するためのインクリメント10、

またはそのままの出力を選択して格納するストアバッファ掃出しポインタ19を含む。

次に、本発明の一実施例の動作を第1図から第3図を参照して詳細に説明する。第1図および第2図を参照すると、サイクル1では主記憶装置8から命令キャッシュI Cacheに命令Aが格納される。命令Aは4ワード命令である。次にサイクル2では主記憶装置8から命令キャッシュI Cacheに命令Bが格納されるとともに命令Aは命令キャッシュI Cacheからセクタ21および命令レジスタIR20を介して加算器AD.ADDERでアドレス生成される。このようにして、命令A、BおよびCの処理は進められる。命令Bは2ワード命令であり、命令Cは1ワード命令である。

第1図から第3図を参照すると、サイクル6で読出データレジスタRDR 5に格納された命令Aに関する第1ワード目の演算結果はサイクル7で線103を介してストアバッファ7の1番地に格納される。

線101を介してIC更新指示が出力されてい

ストアバッファ7から主記憶装置8へのバッファ掃出しを制御するための指示信号を線105に出力するメモリアクセス制御部11、このメモリアクセス制御部11から線105を介して与えられる指示信号に反応してストアバッファ7からの内容を線104を介して主記憶装置8へ送出するレジスタHAR 12、演算制御部1から線101を介して与えられる命令カウンタ更新指示信号に反応して、命令カウンタ更新の場合は“1”を命令終了フラグとして、また命令カウンタ更新でない場合は“0”を線102に出力する命令カウンタ(以下IC)更新制御回路13、命令カウンタ16、IC更新制御回路13から線102を介して与えられる信号に反応して命令カウンタ16からの値をセットするレジスタIC15、IC更新制御回路13からの信号を格納するレジスタWFI 17、このレジスタWFI 17および命令終了フラグバッファWFB 6からの出力のどちらか一方を選択する選択制御回路18、およびこの選択制御回路18からの信号に反応してインクリメント23、デクリメント24、

いため、IC更新制御回路13から線102を介してレジスタWFI 4に格納された値“0”が、上述の命令Aの第1ワード目の演算結果のストアバッファ7への格納動作とともに、命令終了フラグバッファ6の対応するワードに“0”が格納される。

サイクル10で命令カウンタレジスタ15の内容が“A”から“B”に変化するとともに命令終了フラグバッファ4に“1”が格納され、インクリメント23が動作してストアバッファ掃出しポインタ19の内容が“0”から“1”に変化する。これはストアバッファ7内に掃出し可能なデータが1命令あることを示す。

メモリアクセス制御部11はストアバッファ掃出しポインタ19の値が“0”のときはストアバッファ7に掃出し可能なデータがないため、ストアバッファ7から主記憶装置8に対する掃出し動作を抑止する指示信号をレジスタ12に与える。いま、掃出しポインタ19の内容が“1”であるため、メモリアクセス制御部11はストアバッファ7の内容を線104を介して主記憶装置8に供給

するようレジスタ12に指示する。この指示に回答してレジスタ12はストアバッファ7から主記憶装置8に対して命令Aの命令単位の掃出しが開始される。

サイクル10において命令Aの第4ワード目のストアバッファ7への格納が終了しサイクル11において命令Bの第1ワード目のストアバッファ7への格納動作が開始される。

サイクル12において命令カウンタレジスタ15の内容が更新されると、これと同期して命令終了フラグバッファ6に“1”が格納される。この時命令Aの第3ワード目のストアバッファ7から主記憶装置8への掃出し動作が行なわれているため、命令Bの命令単位を主記憶装置8に掃出すことができない。そこでストアバッファ掃出しポインタ19の値は“1”から“2”にカウントアップされる。これは、ストアバッファ内に掃出し可能なストアデータが2命令分存在することを示す。

サイクル13において命令カウンタ16により命令カウンタレジスタ15の内容が“2”から

“3”に更新され、命令Cに関する演算結果が格納されると、これと同期して命令終了フラグバッファ6に値“1”が格納される。この時命令Aの第4ワード目がストアバッファ7から主記憶装置8に対して掃出されており、命令Cに関する演算結果を主記憶装置8に掃出すことはできない。

そこで、ストアバッファ掃出しポインタ19の値は“2”から“3”にカウントアップされ、ストアバッファ7内に掃出し可能なストアデータが3命令分存在することを示している。

サイクル13において、命令Aの第4ワード目の演算結果の主記憶装置8への掃出しとともに命令終了フラグバッファ6から“1”が出力されると、サイクル14においてストアバッファ掃出しポインタ19の内容のデクリメント24によるデクリメントが行なわれ、ポインタ19の内容は“3”から“2”に変化する。この値“2”は、ストアバッファ7内に掃出し可能なストアデータが2命令存在することを示す。

サイクル14において、命令Bの第1ワード目

の演算結果の主記憶装置8への掃出し動作が開始され、サイクル15において、命令Bの第2ワード目の命令単位の主記憶装置8への掃出しが行なわれる。命令終了フラグバッファ6から“1”が検出されるとストアバッファ掃出しポインタ19の値がデクリメント24によりデクリメントされ“1”となる。これは、ストアバッファ7内に掃出し可能なストアデータが1命令分存在していることを示している。

サイクル16において命令Cがストアバッファ7から掃出されると、命令終了フラグバッファ6から“1”が検出され、サイクル17においてストアバッファ掃出しポインタ19の値がデクリメント24によりカウントダウンされ“0”となり、これはストアバッファ7に掃出し可能なデータがないことを示している。

なお、命令終了フラグバッファ6は第3図に示すようなバッファ構造となっており、ストアデータがストアバッファ7に書込まれるタイミングで命令カウンタレジスタ15の値が変更される時に

“1”が、それ以外の時には“0”が書込まれる。また、同一タイミングで命令終了フラグバッファ6にストアされるフラグが“1”ならば、ストアバッファ掃出しポインタ19の値がカウントアップされ、“0”ならばカウントアップされずにその時のポインタ19の値が保持される。

また、ストアバッファ7から主記憶装置8へ掃出すタイミングで命令終了フラグバッファ6の読出動作も行なわれる。読出された命令終了フラグの値が“1”ならば、ストアバッファ掃出しポインタ19の値がカウントダウンされ、命令終了フラグの値が“0”ならばカウントダウンされずにその時の掃出しポインタ19の値が保持される。

次に実行ステージにおける障害検出がなされたときの動作について説明する。例えば、命令Aのサイクル7のEX(実行)ステージにおいて障害が検出されると、命令カウンタ16の更新、命令終了フラグバッファ6へのストア、ストアバッファ掃出しポインタ19の更新、およびストアバッファ7から主記憶装置8への命令Aの掃出しが抑

止され、命令Aからのリトライ（再試行）が指示される。

発明の効果

本発明はストア命令において発行されるストア要求に対して命令単位に命令終了フラグを付加することにより、主記憶装置8の書換えを命令の終了まで送らせる制御を行なうことができるという効果がある。

また、本発明はストアバッファ掃出しポインタ19を設けることにより、ストアバッファ7に格納された最も古い命令単位から主記憶装置8に掃出すことができ、結果としてリトライ率を向上できるという効果がある。

4. 図面の簡単な説明

第1図は本発明の一実施例を示す図、第2図は本発明の一実施例の動作を説明するためのタイムチャート、第3図はストアバッファ、命令終了フラグバッファおよびストアバッファ掃出しポインタの関係を示す図、第4図はパイプライン処理を説明するための図、第5図は従来技術の一例を示

すためのタイムチャートである。

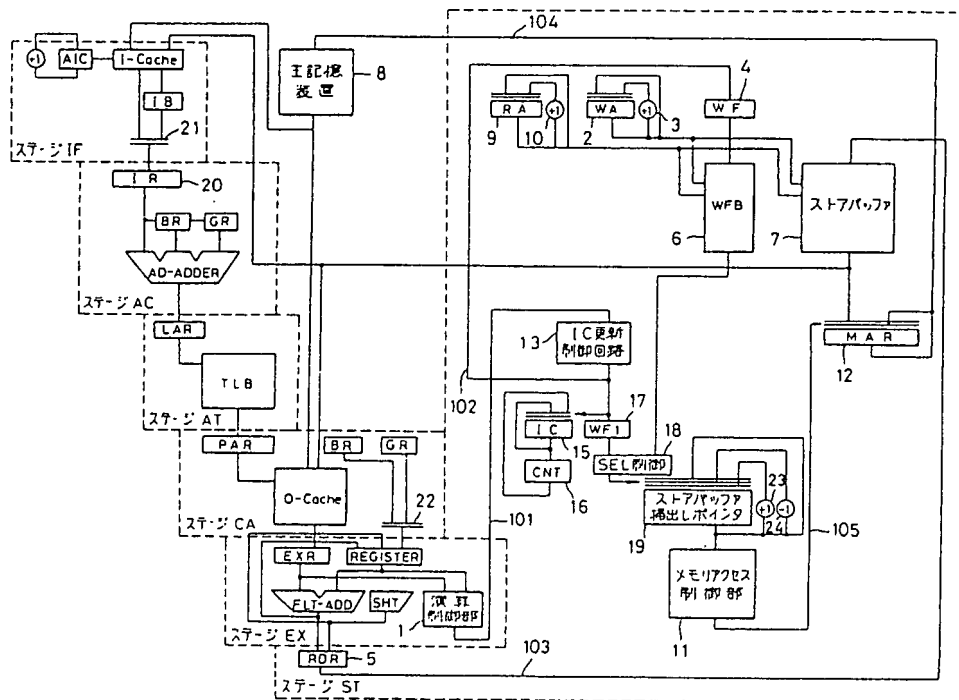
主要部分の符号の説明

- 1 …… 演算制御部
- 2 …… 書込アドレスレジスタ
- 3 …… リードデータレジスタ
- 6 …… 命令終了フラグバッファ
- 7 …… ストアバッファ
- 8 …… 主記憶装置
- 9 …… 読出アドレスレジスタ
- 11 …… メモリアクセス制御部
- 13 …… 命令カウンタ更新制御回路
- 15 …… 命令カウンタレジスタ
- 16 …… 命令カウンタ
- 18 …… 選択制御回路
- 19 …… ストアバッファ掃出しポインタ
- 20 …… 命令レジスタ

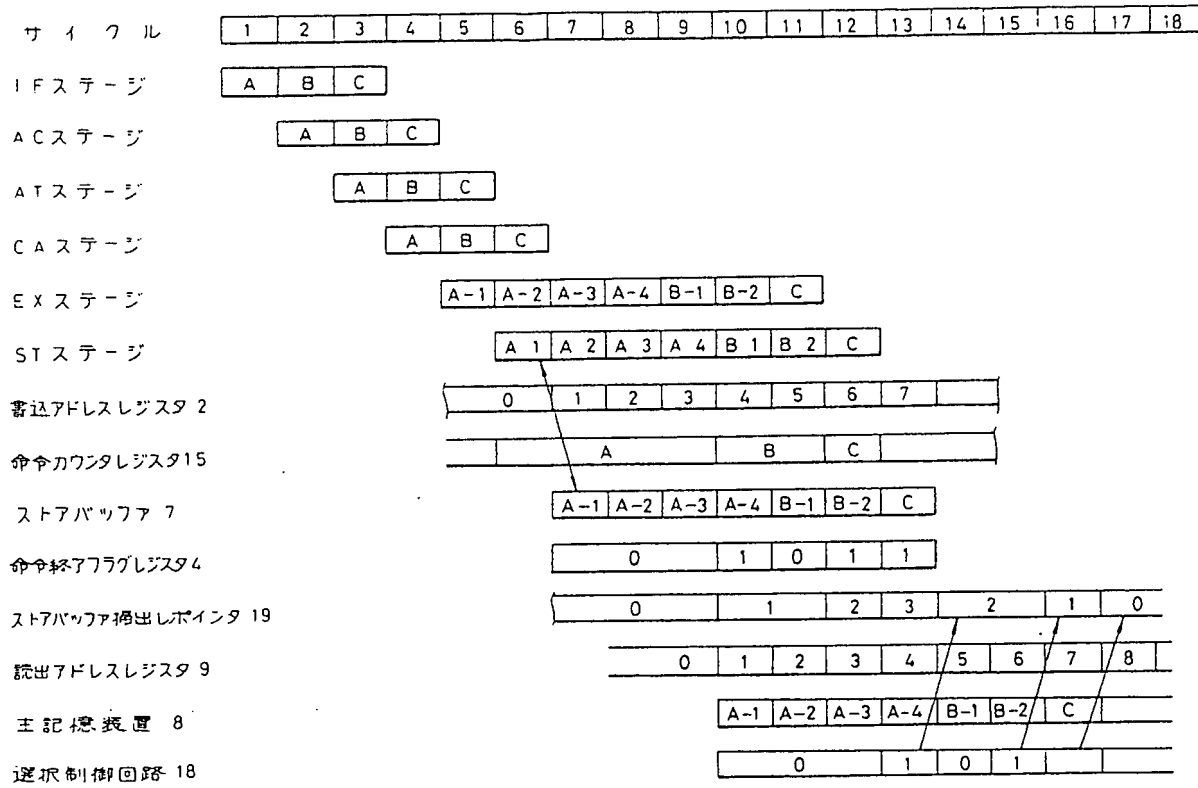
出願人 日本電気株式会社

代理人 弁理士 柳川 信

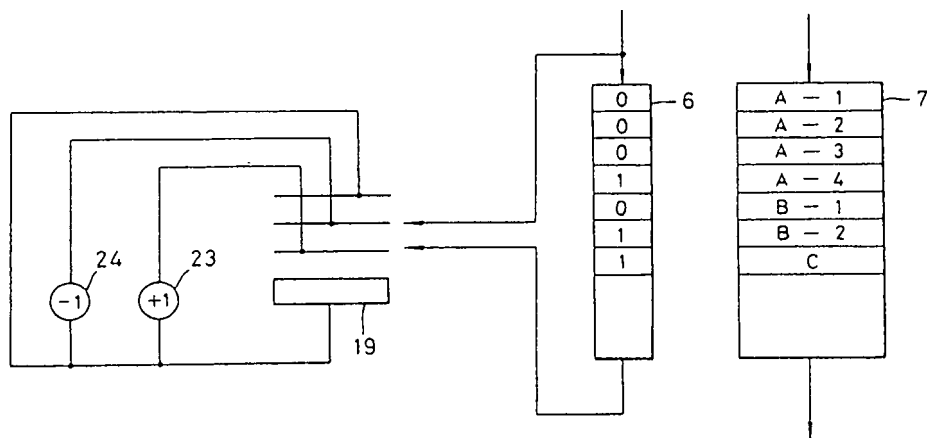
Fig. 1
第1図



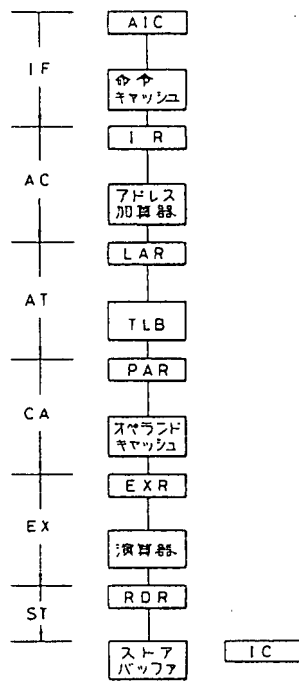
第2図



第3図



第4図



第5図

